The Extension of Universal Generating Function Method to Search for All One-to-Many d-Minimal Paths of Acyclic Multi-State-Arc Flow-Conservation Networks

Wei-Chang Yeh, *Member, IEEE*

***Abstract*—Evaluating network reliability is an important step in the planning, design, and control of systems. The acyclic multi-state-arc flow-conservation network (AMAFCN) is a special multi-state-arc flow-conservation network (MAFCN) of which each arc has independent, discrete, limited multi-states (capacities), and satisfied flow conservation. The AMAFCN is arranged in such a way that no flow leaving a node can return this node through any sequence of nodes, e.g. no cycles exist. For such networks, we are interested in evaluating the AMAFCN reliability, i.e. the probability that the flow from the source node to the sink node is equal to or greater than a demanded flow of d units. A general method for the MAFCN/AMAFCN reliability evaluation is using minimal path/cut vectors with system state d (called the d-MP/d-MC). In this study, we focused on developing an extension of the universal generating function method (UGFM) to find the entire one-to-many d-MP before calculating the AMAFCN reliability between the source node, and some node subset (i.e. one-to-many reliability). The computational complexity of the proposed algorithm is also analyzed. Finally, one example is given to illustrate how entire one-to-many d-MP are generated using the proposed algorithm. Then, all one-to-many reliabilities of this example are calculated.**

***Index Terms*—Acyclic multi-state-arc flow-conservation network, d-MP, one-to-many, reliability, universal generating function method.**

ACRONYM[1]

MNFDN	Multi-state-node flow-disconservation network (without satisfying the conservation law)
AMNFDN	Acyclic MNFDN
MAFCN	Multi-state-arc flow-conservation network which satisfies the conservation law
AMAFCN	Acyclic MAFCN
UGF	Universal Generating Function
UGFM	UGF Method
MP/MC	Minimal path/cut
d-MP/d-MC	d-Minimal path/cut

NOTATION

$\lvert\bullet\rvert$	The number of elements in $\bullet$.
$Min(\bullet)$	The element with the minimal label in the set $\bullet$, e.g. $Min\{-1,4,0\}=-1$.
$G(V,E,W)$	An AMAFCN with the set of nodes $V=\{1,2,\ldots,n\}$, the set of arcs E, the source node 1, and W denotes the max-capacity of arcs. For example, the network in Fig. 1 is an AMAFCN with $V=\{1,2,\ldots,5\}$.
T	The node subset of which there is no arc emanating out from each node in T, e.g. $T=\{4,5\}$ in Fig. 1.
n, m	The number of nodes in V, and arcs in E in $G(V,E,W)$, respectively.
e_{ij}	The arc from nodes i to j in E.
V_i	The node subset $\{1,2,\ldots,i\}\subseteq V$.
i^+	$i^+=\{j\in V-V_i \lvert e_{ij}\in E\}$, e.g. $2^+=\{3,5\}$ in Fig. 1.
V_i^+	$V_i^+=\{j\in V-V_i \lvert e_{kj}\in E$ for $k\in V_i\}$, e.g. $V_3^+=\{4,5\}$ in Fig. 1.
$G(\bullet)=G(\bullet,E,W)$	A sub-network of $G(V,E,W)$, where $\bullet\subseteq V$, e.g. Fig. 2 is $G=G(V_3,E,W)$ of Fig. 1.
$\varepsilon_{i:I}$	$\{e_{ij}\lvert j\in I\neq\varnothing\}\subseteq E$, e.g. $\varepsilon_{2:\{3,5\}}=\{e_{23},e_{25}\}$.
subscript $_{ij\ldots k}$	A simple way to denote the non-empty set $\{i,j,\ldots,k\}$, i.e. $ij\ldots k=\{i,j,\ldots,k\}\subseteq V$, e.g. $\varepsilon_{2:35}=\varepsilon_{2:\{3,5\}}$.
$\varepsilon_{i:I}\varepsilon_{j:J}$	$\varepsilon_{i:I}\varepsilon_{j:J}=\varepsilon_{i:I}\cup\varepsilon_{j:J}$, e.g. $\varepsilon_{1:\{2\}}\varepsilon_{2:\{3,4\}}=\varepsilon_{1:\{2\}}\cup\varepsilon_{2:\{3,4\}}=\{e_{12},e_{23},e_{24}\}$.
$u(i)$	The node-UGF of node i (see the details in Sections II and III).
$U(i)$	The subnet-UGF of node i (see the details in Sections II and III). Note $U(1)=u(1)$.

Manuscript received October 29, 2006; revised March 29, 2007 and June 6, 2007; accepted July 5, 2007. This research was supported in part by the National Science Council of Taiwan, R.O.C., under Grant NSC 95-2221-E-007-180-MY2. Associate Editor: M. Zuo.

The author is with the *e*-Integration&Collaboration Laboratory, Department of Industrial Engineering and Management Engineering, National Tsing Hua University, Hsinchu, Taiwan 300, R.O.C. (e-mail: yeh@ieee.org).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TR.2007.911260

[1]The singular and plural of an acronym are always spelled the same.

$X \otimes Y$	$X \otimes Y = \{\{\alpha \cup \beta\} \mid \text{for all } \alpha \in X \text{ and } \beta \in Y\}$, where $\otimes$ is the composition operator for UGFM (see the details in Sections II and III).
$\underline{\in}$, $\underline{\notin}$	If $B \underline{\in} (\underline{\notin}) \bullet$, then B is (not) the complete coefficient of "z" in $\bullet$, where $\bullet$ is a node-UGF or subnet-UGF. For example, $\varepsilon_{1:2}$, $\varepsilon_{1:3}$, and $\varepsilon_{1:23} \underline{\in} u(1)$ if $u(1) = \varepsilon_{1:2} z + \varepsilon_{1:3} z + \varepsilon_{1:23} z$.
$\omega(\bullet)$	The structure function of $\bullet$.

NOMENCLATURE

Demand level d	The demand level d is a non-negative integer-valued load, or stress requirement for the given problem. Usually it is a random variable whose distribution can be determined through continuous observation, and forecasting.
d-MP candidate	A system-state vector $X = (x_1, x_2, \ldots, x_m)$ is a d-MP candidate iff the maximal flow in $G(V, E, W^*)$ is d, where $W^*(e_i) = x_i$ for all i.
d-MP	A d-MP candidate $X = (x_1, x_2, \ldots, x_m)$ is a d-MP if there is no directed cycle in $G(V, E, W^*)$, where $W^*(e_i) = x_i$ for all i.
$(i : I)$-MP	An union of MP from node i to node subset I. For example, Fig. 2 is a (1:{3})-MP constructed by two MP: $\{e_{12}, e_{23}\}$, and $\{e_{13}\}$.
Reliability	The probability that the required amount of each commodity is transmitted successfully from the source node to the sink node.
Acyclic network	A directed network is acyclic if it contains no directed cycle.
Endpoint	The endpoints of an arc are the vertices that it joins.
State	one of the subsets of the combination of reachable adjacent nodes, e.g. {2}, {3}, and {2,3} are all states of node 1 in Fig. 1.
Node-UGF	The UGF for the basic element of UGFM.
Subnet-UGF	The UGF formed by node-UGF.
Composition Operator	The operator is used to unify node-UGF to subnet-UGF.
One-to-many	Between node 1, and any node subset. For example, one-to-K reliability is the probability that node 1, and all nodes in K are connected for all $K \subseteq V$.
Flow conservation law	The total flow into, and from a node (not source, and target nodes) are all equal.
Topological Numbering	A special way to label the nodes so that if $e_{ij} \in E$, then $i < j$.

ASSUMPTIONS

The AMAFCN satisfies the following assumptions:

1) All nodes are perfectly reliable.
2) All arcs are directed, and failure prone.
3) The arc capacity is an independent, discrete random variable, and takes non-negative integer-values according to a given distribution.
4) No cycle is allowed in the network.
5) The network obeys the conservation law, i.e. the total flows through, into, and from a node (not a source node or a target node) are all equal.

I. INTRODUCTION

In recent years, network reliability theory has been applied extensively to many real-world multi-state networks in which the edges have independent, finite, and integer-valued random capacities such as oil/gas production systems, computer and communication systems, power transmission and distribution systems, transportation systems, etc. Multi-state networks are more practical, and reasonable in many real-life situations than are binary-state systems. Multi-state network reliability analysis thus plays important roles in the network reliability of our modern society [1]–[25].

According to how the method used to transfer the flow (or signal) satisfies the flow conservation law, there are two categories of multi-state network reliability problems: the multi-state-arc flow-conservation network (MAFCN) [1]–[18], and the multi-state-node flow-disconservation network (MNFDN) [19]–[25]. In MAFCN, each arc has a non-negative integer-valued random variable capacity (multi-state-arc), and all flows in the network obey the conservation law [1]–[18]. Conversely, the MNFDN violates the conservation law, and its non-sink node has different states determined by a set of nodes receiving the signal directly from it [19]–[25]. Both have their own applications; for example, electrical power distribution can be modeled as MAFCN [1]–[18], and computer networks or cellular telephone networks be modeled as MNFDN [19]–[25].

The evaluation of the MAFCN, or MNFDN reliability is NP-hard [4], [5]; even the network is acyclic. Many methods had been developed to solve the MAFCN reliability, including the exact methods based on cut/path set [1]–[16], which can calculate the exact value of reliability, and some approximate methods [17], [18], which can estimate the approximate value of network reliability, and avoid the NP-hard problem faced by the cut/path-based methods. Exact network reliability evaluation approaches exploit a variety of tools for system modeling, and reliability index calculation. Among the most popular tools are network-based algorithms founded in terms of either d-MP, or d-MC for the MAFCN/AMAFCN reliability evaluation. A minimal path/cut (MP/MP) is an arc subset, and any proper subset of MC/MP is not a MC/MP, i.e. if any arc is removed from the MC/MP, then the remaining set is no longer a path/cut [5], [6], [11], [12], [16], [24]. The d-MP/d-MC is a MP/MC vector such that the maximal flow (minimal cut) from the

source node to the sink node (with no cycle) via this vector is d [7]–[10], [13]–[16].

The general method for reliability evaluation limits the discussions to two-terminal reliability analysis. To the author's best knowledge, most of the existing-known algorithms for finding all d-MP need to search for all MP first. Unfortunately, the search for all MP or MC is also NP-hard [5], [6], [11], [12], [18], [24]. Recently, Yeh proposed a novel algorithm to search for d-MP without knowing the MP in advance based on a special mathematical programming model for the flow [15]. Yeh's algorithm is easy to understand, and implement. However, Yeh's algorithm works only for the one-to-one d-MP problem [15]. It seems to be superfluous when applied to the one-to-many AMAFCN reliability problem. The need for an efficient method to search for all one-to-many d-MP in the AMAFCN thus arises.

UGFM was introduced in [19], and was proven to be very effective for evaluating the reliability of different types of multi-state networks [20], [21], [23], [25], because it does not require a great computational effort. The further developments and applications of the UGF technique were presented in [21], [22], [25]; and detailed descriptions are available in [21], [22] that summarized recent achievements in the field. The UGFM is straightforward, effective, and universal [20]. It involves intuitively simple recursive procedures combined with simplification techniques.

The purpose of our paper is to extend UGFM using the simple operation of sets to search for entire one-to-many d-MP to calculate the one-to-many AMAFCN reliability.

This paper is organized as follows. The traditional UGFM is introduced briefly in Section II. Section III contains the discussions on the formulation of the proposed UGFM with its important properties and characteristics. Section IV presents the algorithm of the proposed UGFM in detail, and its computational complexity is also analyzed. Section V illustrates how the proposed UGFM generates entire one-to-many d-MP, and evaluates the AMAFCN reliability after obtaining d-MP using three numerical bench examples. Concluding remarks are given in Section VI.

II. UGFM: Main Definitions and Properties

The detailed description of the UGFM is presented in [20], [21]. In this section, a brief introduction to the traditional UGFM for the AMNFDN is given. To assure that all states are correct, and completely included in the UGFM, we define two types of UGF: the individual UGF (called the node-UGF in this study), and the group UGF (called the subnet-UGF). The node-UGF is the core of UGFM. The node-UGF of node i represents all the possible states of the node i by the associated probability of each state. The node-UGF of node i is defined as a polynomial

$$u(i) = \sum_{k=1}^{K_i} q_{i,k} z^{x_{i,k}} \tag{1}$$

where node i has K_i possible different states, $q_{i,k}$ is the associated probability that the state of node i is equal to $x_{i,k}$ (the node's performance), and $\sum_{k=1}^{K_i} q_{i,k} = 1$.

To obtain the UGF of a connected sub-network $G(V_i) = G(V_i, E, W)$, a general composition operator $\otimes$ is used over node-UGF $u(1)$, $u(2)$,..., and $u(i)$ represents the sub-network $G(V_i)$:

$$U(i) = u(1) \otimes u(2) \otimes \ldots \otimes u(i) \tag{2}$$

$$= \sum_{k=1}^{K_1} q_{1,k} z^{x_{1,k}} \otimes \sum_{k=1}^{K_2} q_{2,k} z^{x_{2,k}} \otimes \ldots \otimes \sum_{k=1}^{K_i} q_{i,k} z^{x_{i,k}} \tag{3}$$

$$= \sum_{k_1=1}^{K_1} \sum_{k_2=1}^{K_2} \cdots \sum_{k_i=1}^{K_i} q_{1,k_1} q_{2,k_2} \ldots q_{i,k_i} z^{\omega(x_{1,k_1}, x_{2,k_2}, \ldots, x_{i,k_i})} \tag{4}$$

where $U(i)$ is the subnet-UGF representation of the sub-network $G(V_i)$, and $U(1) = u(1)$. The definition of the structure function $\omega(\bullet)$ strictly depends on the physical nature of the sub-network (e.g. performance measure, reliability, etc.), and on the nature of the interaction of nodes (e.g. the connection).

There is a fundamental assumption with using UGFM: the nodes are numbered in such a way that if $e_{ij} \in E$, then $i < j$, i.e. node i cannot receive the signal (flow) from any node j if $i > j$. The sequence of finding the subnet-UGF must follow the same order of the node labeled in the existing UGFM, i.e. $U(1), U(2), \ldots, U(|n|)$. All new elements in $U(i)$ are obtained from an element in $U(i-1)$, and an element in $u(i)$. Hence, one can obtain a recursive expression for the subnet-UGF $U(i)$, and consecutively applying

$$U(i) = u_i(i) \otimes U(i-1). \tag{5}$$

Equation (5) represents the essence of the universal generating function (UGF) technique, which is based on using a universal z-transform, and composition operators. The subnet-UGF unifies the node-UGF in a recursive way using some special operators, e.g. the composition operator here. A new node-UGF is merged into the current sub-network each time until all possible node-UGF are in the final subnet-UGF. Both different kinds of UGF record all their own states. We can find the corresponding answer (e.g. the d-MC/d-MP) in the subnet-UGF if all states are found correctly, and completely.

III. UGFM Formulation, and Preliminaries

This section presents the proposed UGFM for finding one-to-many d-MP in AMAFCN. A brief introduction to the basic elements, and the fundamental concept of the proposed UGFM with some useful properties are given here first. From the above introduction, the key point to successful implementation of UGFM is that each state of nodes, and sub-networks must be correct, and completely included in node-UGF, and subnet-UGF. The UGFM is a complete examination of all possible states of each node-/subnet-UGF, which implies its correctness. In the proposed UGFM, both node-UGF $u(i)$, and subnet-UGF $U(i)$ are special sets of arc subsets. All arcs in the former set of subsets included the same tail: node i; each

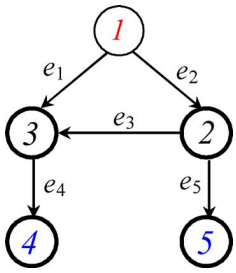

Fig. 1. An AMAFCN.

element in the latter is a $(i : V_i^+)$-MP. Before the formal introduction of the proposed node-UGF, the proposed node-UGF is defined as

$$u(i) = \sum_{I \neq \varnothing \text{ and } I \subseteq V_i^+} \varepsilon_{i:I} z \tag{6}$$

where $I \subseteq V_i^+$ is a non-empty node subset emanating from node i, and $\varepsilon_{i:I}$ is the arc subset s.t. the node i can send flows to all nodes in I via all arcs in $\varepsilon_{i:I}$,. For example, in Fig. 1, we have $u(1)$, and $u(2)$ as

$$\begin{aligned} u(1) &= \sum_{I \neq \varnothing \text{ and } I \subseteq V_1^+ = \{2,3\}} \varepsilon_{1:I} z \\ &= \varepsilon_{1:2} z + \varepsilon_{1:3} z + \varepsilon_{1:23} z \qquad (7) \\ &= U(1), \qquad (8) \end{aligned}$$

and

$$\begin{aligned} u(2) &= \sum_{I \neq \varnothing \text{ and } I \subseteq V_2^+ = \{3,5\}} \varepsilon_{2:I} z \qquad (9) \\ &= \varepsilon_{2:3} z + \varepsilon_{2:5} z + \varepsilon_{2:3,5} z \qquad (10) \end{aligned}$$

The following simple observation discusses an important, special characteristic of one-to-many MP.

Theorem 1: Each $(1 : K_J)$-MP is a union of one $(1 : J)$-MP, and some $(j : K_k)$-MP if $j \in J^+$, $\bigcup_k K_k = K_J$, and

$$\begin{cases} Indegree(i) = 0 & \text{if } i = 1 \\ Indegree(i) > 0 & \text{if } i \neq 1 \end{cases}, \tag{11}$$

$$\begin{cases} Outdegree(i) = 0 & \text{if } i \in K_J \\ Outdegree(i) > 0 & \text{if } i \notin K_J \end{cases}, \tag{12}$$

where i is an endpoint in $(1 : K_J)$-MP.

The following rules are emerged in the proposed UGFM to record all $(1 : V_i^+)$-MP in $G(V_i^+)$.

Rule 1: If Bz_j, and $\varepsilon_{i:I} z$ are independent terms in $U(i-1)$, and $u(i)$, respectively; then,

$$Bz_J \otimes \varepsilon_{i:I} z = \begin{cases} Bz_J, & \text{if } i \notin J \\ B\varepsilon_{i:I} z_{I \cap J - \{i\}} & \text{otherwise} \end{cases}. \tag{13}$$

Rule 2: The function $u(i)$ indicates all possible ways to transfer flow from node i. For example, Fig. 3 shows the corresponding sub-networks w.r.t. to the coefficients in $u(1)$ in (7). The node labels are numbered such that $u < v$ if arc $e_{uv} \in E$.

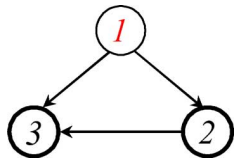

Fig. 2. $G(V_3, E, W)$ of Fig. 1.

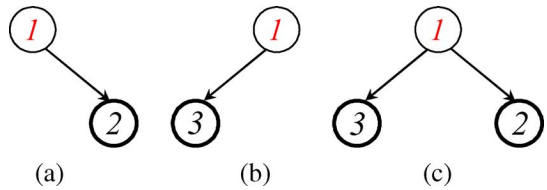

Fig. 3. The coefficients in $u(1)$: (a) $\varepsilon_{1:2} = \{e_{12}\}$; (b) $\varepsilon_{1:3} = \{e_{13}\}$; (c) $\varepsilon_{1:\{2,3\}} = \{e_{12}, e_{13}\}$.

Rule 1 is to assure that all newly obtained $(1{:}V_i^+)$-MP are connected. Rule 2 is analogous to the necessary assumption for the AMNFDN reliability problem, i.e. AMAFCN is acyclic, and its nodes can be topologically numbered such that $u < v$ if arc $e_{uv} \in E$. [5]. The composition operator $\otimes$ is similar to the union operator. It plays a central role in UGFM. The definition of $U(i)$ is a recursive expression based on a composition operator $\otimes$ in the following manner:

$$\begin{aligned} U(i) &= U(i-1) \otimes u(i) \qquad (14) \\ &= \{\{B \cup b\} = Bb | B \text{ and } b \text{ are connected}\} \\ &\quad \cup \{B | B \text{ and } b \text{ are disconnected}\} \qquad (15) \end{aligned}$$

where for all $i = 2, 3, \ldots, n-1$, $B \underline{\in} U(i-1)$, $b \underline{\in} u(i)$, and

$$U(1) = u(1). \tag{16}$$

Corollary 1: If $B \underline{\in} U(j)$, then B is a connected arc subset, and

$$\begin{cases} Indegree(i) = 0 & \text{if } i = 1 \\ Indegree(i) > 0 & \text{if } i \neq 1 \end{cases}, \tag{17}$$

$$\begin{cases} Outdegree(i) = 0 & \text{if } i \in j^+ \\ Outdegree(i) > 0 & \text{if } i \notin j^+ \end{cases}, \tag{18}$$

where i is an endpoint in B.

From (12), $U(i-1)$ is extended, and enlarged to $U(i)$ using the composition operator $\otimes$. For example, also in Fig. 1, we have

$$\begin{aligned} U(2) &= U(1) \otimes u(2) \qquad (19) \\ &= u(1) \otimes u(2) \qquad (20) \\ &= (\varepsilon_{1:\{2\}} z + \varepsilon_{1:\{3\}} z + \varepsilon_{1:\{2,3\}} z) \\ &\quad \otimes (\varepsilon_{2:\{3\}} z + \varepsilon_{2:\{5\}} z + \varepsilon_{2:\{3,5\}} z) \qquad (21) \\ &= \varepsilon_{1:\{3\}} z_3 + \left[(\varepsilon_{1:\{2\}} z_2 + \varepsilon_{1:\{2,3\}} z_{\{2,3\}}) \right. \\ &\quad \left. \otimes (\varepsilon_{2:\{3\}} z + \varepsilon_{2:\{5\}} z + \varepsilon_{2:\{3,5\}} z)\right] \qquad (22) \\ &= \varepsilon_{1:\{3\}} z_{\{3\}} + \varepsilon_{1:\{2\}} \varepsilon_{2:\{3\}} z_{\{3\}} + \varepsilon_{1:\{2\}} \varepsilon_{2:\{5\}} z_{\{5\}} \\ &\quad + \varepsilon_{1:2} \varepsilon_{2:\{3,5\}} z_{\{3,5\}} + \varepsilon_{1:\{2,3\}} \varepsilon_{2:\{3\}} z_{\{3\}} \\ &\quad + \varepsilon_{1:\{2,3\}} \varepsilon_{2:\{5\}} z_{\{3,5\}} + \varepsilon_{1:\{2,3\}} \varepsilon_{2:\{3,5\}} z_{\{3,5\}}. \qquad (23) \end{aligned}$$

Immediately from the above rules, some elementary facts concerning the node labels, and the coefficients of node-UGF and subnet-UGF are all brought together as follows.

Property 1: If there is a directed path from nodes i to j, then $i < j$. Moreover, there is a directed path from nodes i to j iff $i < j$, and $i \notin T$.

Property 2: $\mathrm{q}\varepsilon_{i:I} \underline{\in} u(i)$ iff $\{e_{ij} |$ for all $j \in I\} \subseteq E$.

Property 3: $\varepsilon_{qqi:I} B \underline{\in} U(i)$ iff $\varepsilon_{i:I} \underline{\in} u(i)$, $B \underline{\in} U(i-1)$, and $B \cup \varepsilon_{i:I}$ is a connected arc subset.

Property 4: $B \underline{\in} U(i)$, and node i is not an endpoint in B iff $B \underline{\in} U(i-1)$, and $B \cup \varepsilon_{i:I}$ is a disconnected arc subset for all $\varepsilon_{i:I} \underline{\in} u(i)$.

From Corollary 1, Properties 1–4, and Theorem 1, we obtain the following fundamental result.

Theorem 2: The following statements are equivalent.

(a)

$$B \underline{\in} U(i). \tag{24}$$

(b)

$$B \text{ is a } (1 : V_i^+) - \text{MP}. \tag{25}$$

(c) B is a connected arc subset, i.e. $G(\{$the endpoints of the arcs in $B\}, B)$ is a connected sub-network.

(d) $B = \varepsilon_{i:I} B^*$, where $\varepsilon_{i:I} \underline{\in} u(i)$, and $B^* \underline{\in} U(i-1)$.

Theorem 2 summarizes the preceding discussion about the relationships among the coefficients of subnet-UGF $U(i)$, (1:V_i^+)-MP, and the corresponding sub-network of $U(i)$. Moreover, it shows that all (1:V_i^+)-MP are correct and completely included in $U(i)$, and $U(i)$ contains only all (1:V_i^+)-MP. The following results are easy, immediate consequences of the preceding theorem.

Corollary 2:

$$\{\tau | \text{ for all } (1 : V_i^+) - \text{MP } \tau\} = \{B | \text{ for all } B \underline{\in} U(i)\}. \tag{26}$$

Corollary 3: All (1:I)-MP are included in $U(i) = U(i-1) \otimes u(i)$ for all $I \subseteq V_i^+$, and i.

There is no arc emanating out from each node in T, we have $u(i) = \varnothing$ for $i \in T$. Therefore, it is easy to show that the procedure to calculate all subnet-UGF terminates before including any target node. Therefore, we have established the follow result.

Theorem 3: All (1:V_i^+)-MP are found after calculating $U(|V - T|)$ for all i.

From Theorem 3, all (1:●)-MP are found after implementing (14) repeatedly until $u(|V - T|)$ is included. The next step is to transfer each (1:V_i^+)-MP to one-to-many d-M for all i. Each (1:V_i^+)-MP forms a sub-network. For example, $\varepsilon_{1:\{3\}} = \{e_{13}\}$, $\varepsilon_{1:\{2\}}\varepsilon_{2:\{3\}} = \{e_{12}, e_{23}\}$, $\varepsilon_{1:\{2\}}\varepsilon_{2:\{5\}} = \{e_{12}, e_{25}\}$, $\varepsilon_{1:2}\varepsilon_{2:\{3,5\}} = \{e_{12}, e_{23}, e_{25}\}$, $\varepsilon_{1:\{2,3\}}\varepsilon_{2:\{3\}} = \{e_{12}, e_{13}, e_{23}\}$, $\varepsilon_{1:\{2,3\}}\varepsilon_{2:\{5\}} = \{e_{12}, e_{13}, e_{25}\}$, and $\varepsilon_{1:\{2,3\}}\varepsilon_{2:\{3,5\}} = \{e_{12}, e_{13}, e_{23}, e_{25}\}$ are (1:V_2^+)-MP in $U(2)$; and their corresponding connected sub-networks from node 1 to nodes in $V_2^+ = \{3, 5\}$ in Fig. 1 are shown in Fig. 4, respectively.

A (1:V_i^+)-MP, say B, is a d-MP if there are exactly d units of flow which can be sent from node 1 to V_i^+ via all arcs in B. Therefore, the key to transfer B into one-to-many d-MP is to send d units of flow from nodes 1 to V_i^+ via all arcs in B.

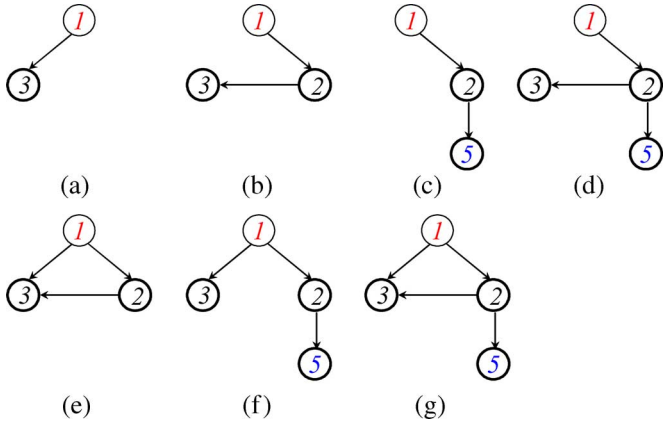

Fig. 4. The corresponding sub-networks w.r.t. the $(1 : V_2^+) - \text{MP}$ in $U(2)$. (a) $\varepsilon_{1:\{3\}}$; (b) $\varepsilon_{1:\{2\}}\varepsilon_{2:\{3\}}$; (c) $\varepsilon_{1:\{2\}}\varepsilon_{2:\{5\}}$; (d) $\varepsilon_{1:2}\varepsilon_{2:\{3,5\}}$; (e) $\varepsilon_{1:\{2,3\}}\varepsilon_{2:\{3\}}$; (f) $\varepsilon_{1:\{2,3\}}\varepsilon_{2:\{5\}}$; (g) $\varepsilon_{1:\{2,3\}}\varepsilon_{2:\{3,5\}}$.

The max-flow problem is the most fundamental of all network problems, and it can be formulated as follows:

$$\text{Maximize} \quad f \tag{27}$$

$$\text{such that} \quad \sum_{j=1}^{n} x_{ij} = \sum_{k=1}^{n} x_{ki},$$

$$\text{for all } x_{ij}, x_{ki} \in E, i \neq 1 \text{ or } n \tag{28}$$

$$\sum_{i=2}^{n} x_{1i} = \sum_{j=1}^{n-1} x_{jn} = f,$$

$$\text{for all } x_{1i}, x_{jn} \in E \tag{29}$$

$$0 \leq x_{ij} \leq W(e_{ij}) \quad \text{for all } e_{ij} \in E \tag{30}$$

where f represents the amount of flow in the network from node 1 to node n. The preceding linear programming model is adapted into a special integer programming model to find all one-to-many d-MP candidates using the implicit enumeration technique. It is based on the definition of the d-MP candidate, and the above max-flow mathematical programming model.

Theorem 4: A feasible solution, say $X = (x_1, x_2, \ldots, x_m)$, of the following mathematical programming model with respect to (1:V_i^+)-MP Δ is a d-MP candidate:

$$\sum_{j=1}^{n} x_{ij} = \sum_{k=1}^{n} x_{ki}, \quad \text{for all } e_{ij}, e_{ki} \in \Delta,$$

$$\text{and } i \notin 1 \cup V_i^+; \tag{31}$$

$$\sum_{i=2}^{n} x_{1i} = \sum_{j=1}^{n-1} x_{jl} = d, \quad \text{for all } e_{1i}, e_{jl} \in \Delta,$$

$$\text{and } l \in V_i^+; \tag{32}$$

$$x_{ij} \leq Min\{d, W(e_{ij})\} \quad \text{for all } e_{ij} \in \Delta; \tag{33}$$

$$1 \leq x_{ij} \quad \text{for all } e_{ij} \in \Delta. \tag{34}$$

Proof: The proof follows from the definition of the one-to-many d-MP candidate, and (28)–(30).

Theorem 4 is based on the flow conversation law (31), the definition of d-MP candidates (i.e. the maximal flow for each one-to-many d-MP candidate is d) (32), the capacity limitation

of each arc (33), and the requirement that the flow needs to go through all arcs in (1:V_i^+)-MP (34). Obviously, all one-to-many d-MP are one-to-many d-MP candidates. The following lemma is an efficient tool to verify whether a d-MP candidate is a real d-MP. It is first proposed, and proved in [16]. It is also true for one-to-many d-MP candidates.

Property 5: A d-MP candidate X is a d-MP if there is no directed cycle in X.

If the flow network is acyclic, then there is no directed cycle in each one-to-many d-MP candidate, i.e. each one-to-many d-MP candidate is a real one-to-many d-MP by Property 5. Therefore, we have the following theorem immediately.

Theorem 5: If the network is acyclic, then every one-to-many d-MP candidate is a real one-to-many d-MP.

Note that all the theory that was introduced in this study is correct only for the case where performance is considered in the sense of capacity (i.e. a flow transmission system). When performance is considered, as for example the data processing speed, the paper's results are not correct.

IV. The Proposed UGFM

The proposed UGFM maintains connected MP sets in the corresponding subnet UGF at each iteration, and successively enlarges & lengthens them into another connected MP set until all non-target node-UGF are included. In the following discussion, we described in greater detail how the proposed algorithm uses the UGFM to search for all one-to-many d-MP. This discussion highlights the UGF in designing an efficient, simple implementation of the algorithm.

Algorithm: Find all one-to-many d-MP between the source node, and all nodes in I using the AMAFCN for all $I \subseteq V$.

Input: An AMAFCN $G(V, E, W)$ with the source node s, and source node set T.

Output: All one-to-many d-MP in the AMAFCN.

STEP 0. Construct $u(1)$, let $U(1) = u(1)$, $i = 1$, and go to STEP 2.

STEP 1. Calculate $U(i) = U(i-1) \otimes u(i)$ which is defined in Section II.

STEP 2. Let $X_J = \{\text{all } (1 : J) - \text{MP,i.e. the coefficient of } z_J \text{ in } U(i)\}$ for all $J \subseteq V_i^+$.

STEP 3. If $i < n - |T|$, let $i = i + 1$, and go to STEP 1.

STEP 4. Construct the mathematical model listed in (31)–(34) to each (1:I)-MP for all $I \subseteq V$; and use the implicit algorithm to solve this model to find all one-to-many d-MP between node 1, and node set I.

Basically, the proposed UGFM is a complete examination of all possible states of each node. The correctness of the above procedure follows from the definition of node-/subnet-UGF, Theorems 1–5, and Properties 1–5. To analyze the time complexity, we make several observations. Notice that the major time complexity of the above proposed UGFM is induced from STEP 1, and the number of states in $u(i)$ is $O(2^{outdegee(i)})$. Therefore, we have

$$\prod_{i \notin T} O(2^{outdegree(i)}) = O\left(2^{\sum_{i \notin T} outdegree(i)}\right). \tag{35}$$

Besides, $outdegree(i) = 0$ for all $i \in T$, and the total outdegree of all nodes is $|E|$ form the following property.

Property 6: (the First Theorem of Graph Theory)

$$\sum_{i \in V} outdegree(i) = |E|. \tag{36}$$

Each state in $U(i)$ needs to be recorded, and replaced by a new state after calculating $U(i+1)$. The total number of states in $U(|V - T|)$ is also equal to (36). Thus, we have therefore constructively proved the following result.

Theorem 6: The proposed UGFM needs $O(2^{|E|})$ time complexity to find all one-to-many d-MP, and $O(2^{|E|})$ units of memory capacity to save all one-to-many d-MP.

V. Illustrative Examples

The general procedure is best illustrated with an example. The total degree of all nodes will grow exponentially with the number of nodes in the worst case [14], [15]. The AMAFCN reliability evaluation possesses a computational difficulty that grows exponentially with the network size. Owing to this inherent problem, instead of presenting practically large network systems, a benchmark network shown in Fig. 1 is selected to demonstrate this methodology as in all existing algorithms [20]–[25].

The first example illustrates Steps 0-3 of the proposed UGFM for finding entire (1:I)-MP for all $I \subseteq V$. These (1:I)-MP are all transferred into one-to-many 2-MP in Example 2 to solve the one-to-many d-MP problem for $d = 2$ using Step 4 of the proposed UGFM.

Example 1: Consider the benchmark AMAFCN presented in Fig. 1, where node 1 is the source node. Steps 0-3 of the proposed UGFM are used to search the entire (1:I)-MP for all $I \subseteq V$.

Solution:

STEP 0. Let $i = 1$, and set up

$$U(1) = u(1) \tag{37}$$

$$= \varepsilon_{1:\{2\}} z + \varepsilon_{1:\{3\}} z + \varepsilon_{1:\{2,3\}} z \tag{38}$$

STEP 1. Because $i = 1 < n - |T| = 3$, let $i = i + 1 = 2$, and let $X_2 = \{\varepsilon_{1:\{2\}}\}$.

STEP 2. Set up $u(2)$ as

$$u(2) = \varepsilon_{2:\{3\}} z + \varepsilon_{2:\{5\}} z + \varepsilon_{2:\{3,5\}} z \tag{39}$$

STEP 3. Calculate the subnet-UGF $U(2) = U(1) \otimes u(2)$ as follows (19)–(23):

$$U(2) = U(1) \otimes u(2) \tag{40}$$

$$= (\varepsilon_{1:\{2\}} z + \varepsilon_{1:\{3\}} z + \varepsilon_{1:\{2,3\}} z) \otimes (\varepsilon_{2:\{3\}} z + \varepsilon_{2:\{5\}} z + \varepsilon_{2:\{3,5\}} z) \quad (41)$$

$$= \varepsilon_{1:\{3\}} z_{\{3\}} + \varepsilon_{1:\{2\}}\varepsilon_{2:\{3\}} z_{\{3\}} + \varepsilon_{1:\{2\}}\varepsilon_{2:\{5\}} z_{\{5\}} + \varepsilon_{1:2}\varepsilon_{2:\{3,5\}} z_{\{3,5\}} + \varepsilon_{1:\{2,3\}}\varepsilon_{2:\{3\}} z_{\{3\}} + \varepsilon_{1:\{2,3\}}\varepsilon_{2:\{5\}} z_{\{3,5\}} + \varepsilon_{1:\{2,3\}}\varepsilon_{2:\{3,5\}} z_{\{3,5\}} \quad (42)$$

$$= \left[\varepsilon_{1:\{3\}} + \varepsilon_{1:\{2\}}\varepsilon_{2:\{3\}} + \varepsilon_{1:\{2,3\}}\varepsilon_{2:\{3\}}\right] z_{\{3\}} + \left[\varepsilon_{1:2}\varepsilon_{2:\{3,5\}} + \varepsilon_{1:\{2,3\}}\varepsilon_{2:\{5\}} + \varepsilon_{1:\{2,3\}}\varepsilon_{2:\{3,5\}}\right] z_{\{3,5\}} + \left[\varepsilon_{1:\{2\}}\varepsilon_{2:\{5\}}\right] z_{\{5\}} \quad (43)$$

STEP 1. Because $i = 2 < n - |T| = 3$, let $i = i + 1 = 3$, and let

$$X_{1:\{3\}} = \varepsilon_{1:\{3\}} + \varepsilon_{1:\{2\}}\varepsilon_{2:\{3\}} + \varepsilon_{1:\{2,3\}}\varepsilon_{2:\{3\}}, \quad (44)$$

and

$$X_{1:\{3,5\}} = \varepsilon_{1:2}\varepsilon_{2:\{3,5\}} + \varepsilon_{1:\{2,3\}}\varepsilon_{2:\{5\}} + \varepsilon_{1:\{2,3\}}\varepsilon_{2:\{3,5\}}. \quad (45)$$

STEP 2. Set up $u(3)$ as

$$u(3) = \varepsilon_{3:\{4\}} z. \quad (46)$$

STEP 3. Calculate the subnet-UGF $U(3) = U(2) \otimes u(3)$ as

$$U(3) = U(2) \otimes u(3) \quad (47)$$

$$= (\varepsilon_{1:\{3\}} z_{\{3\}} + \varepsilon_{1:\{2\}}\varepsilon_{2:\{3\}} z_{\{3\}} + \varepsilon_{1:\{2\}}\varepsilon_{2:\{5\}} z_{\{5\}} + \varepsilon_{1:2}\varepsilon_{2:\{3,5\}} z_{\{3,5\}} + \varepsilon_{1:\{2,3\}}\varepsilon_{2:\{3\}} z_{\{3\}} + \varepsilon_{1:\{2,3\}}\varepsilon_{2:\{5\}} z_{\{3,5\}} + \varepsilon_{1:\{2,3\}}\varepsilon_{2:\{3,5\}} z_{\{3,5\}}) \otimes (\varepsilon_{3:\{4\}} z) \quad (48)$$

$$= \varepsilon_{1:3}\varepsilon_{3:\{4\}} z_{\{4\}} + \varepsilon_{1:\{2\}}\varepsilon_{2:\{3\}}\varepsilon_{3:\{4\}} z_{\{4\}} + \varepsilon_{1:\{2\}}\varepsilon_{2:\{5\}} z_{\{5\}} + \varepsilon_{1:2}\varepsilon_{2:\{3,5\}}\varepsilon_{3:\{4\}} z_{\{4,5\}} + \varepsilon_{1:\{2,3\}}\varepsilon_{2:\{3\}}\varepsilon_{3:\{4\}} z_{\{4\}} + \varepsilon_{1:\{2,3\}}\varepsilon_{2:\{5\}}\varepsilon_{3:\{4\}} z_{\{4,5\}} + \varepsilon_{1:\{2,3\}}\varepsilon_{2:\{3,5\}}\varepsilon_{3:\{4\}} z_{\{4,5\}}) \quad (49)$$

$$= \left[\varepsilon_{1:\{3\}}\varepsilon_{3:\{4\}} + \varepsilon_{1:\{2\}}\varepsilon_{2:\{3\}}\varepsilon_{3:\{4\}} + \varepsilon_{1:\{2,3\}}\varepsilon_{2:\{3\}}\varepsilon_{3:\{4\}}\right] z_{\{4\}} + \left[\varepsilon_{1:2}\varepsilon_{2:\{3,5\}}\varepsilon_{3:\{4\}} + \varepsilon_{1:\{2,3\}}\varepsilon_{2:\{5\}}\varepsilon_{3:\{4\}} + \varepsilon_{1:\{2,3\}}\varepsilon_{2:\{3,5\}}\varepsilon_{3:\{4\}}\right] z_{\{4,5\}} + \left[\varepsilon_{1:\{2\}}\varepsilon_{2:\{5\}}\right] z_{\{5\}} \quad (50)$$

STEP 1. Because $i = n - |T| = 3$, go to STEP 4.
STEP 4. Let

$$X_{1:\{4\}} = \varepsilon_{1:\{3\}}\varepsilon_{3:\{4\}} + \varepsilon_{1:\{2\}}\varepsilon_{2:\{3\}}\varepsilon_{3:\{4\}} + \varepsilon_{1:\{2,3\}}\varepsilon_{2:\{3\}}\varepsilon_{3:\{4\}}, \quad (51)$$

$$X_{1:\{4,5\}} = \varepsilon_{1:2}\varepsilon_{2:\{3,5\}}\varepsilon_{3:\{4\}} + \varepsilon_{1:\{2,3\}}\varepsilon_{2:\{5\}}\varepsilon_{3:\{4\}} + \varepsilon_{1:\{2,3\}}\varepsilon_{2:\{3,5\}}\varepsilon_{3:\{4\}}, \quad (52)$$

$$X_{1:\{5\}} = \varepsilon_{1:\{2\}}\varepsilon_{2:\{5\}}, \quad (53)$$

and halt. □

TABLE I
THE PROBABILITY DISTRIBUTIONS OF THE ARC CAPACITIES OF FIG. 1

i	e_i	$W(e_i)$	$Pr(W(e_i))$
1	e_1=e_{13}	2	.7
		1	.2
		0	.1
2	e_2=e_{12}	1	.8
		0	.2
3	e_3=e_{23}	3	.4
		2	.3
		1	.2
		0	.1
4	e_4=e_{34}	3	.5
		2	.2
		1	.2
		0	.1
5	e_5=e_{25}	2	.8
		1	.1
		0	.1

The result for finding all of $(1{:}I)$-MP for $I \subseteq V$, and the corresponding node-/subnet-UGF is listed in the first five columns in Table II.

Example 2: The probability distributions of the arc capacities of Fig. 1 are listed in Table I. Find all one-to-many 2-MP from node 1 to node set I in Fig. 1 using obtained $(1{,}\{I\})$-MP in Example 1, where all $I \subseteq V$.

Solution: We only demonstrate the procedure to transfer $(1{:}\{4\})$-MP into one-to-$\{4\}$ 2-MP in the following. The results of all 2-MP are listed in the second column from the right in Table II.

1) The mathematical programming formulation for $\varepsilon_{1:\{3\}}\varepsilon_{3:\{4\}}$ is

$$C(e_{13}) = C(e_{34}) = 2, \quad (54)$$
$$1 \le C(e_{13}) \le W(e_{13}) = 2, \quad \text{and} \quad (55)$$
$$1 \le C(e_{34}) \le W(e_{34}) = 3. \quad (56)$$

Therefore, $\mathrm{X} = (C(e_{13}), C(e_{12}), C(e_{23}), C(e_{34}), C(e_{25})) = (2,0,0,2,0)$ is a 2-MP from nodes 1 to 4.

2) The mathematical programming formulation for $\varepsilon_{1:\{2\}}\varepsilon_{2:\{3\}}\varepsilon_{3:\{4\}}$ is

$$C(e_{12}) = C(e_{23}) = C(e_{34}) = 2, \quad (57)$$
$$1 \le C(e_{12}) \le W(e_{12}) = 1, \quad (58)$$
$$1 \le C(e_{23}) \le W(e_{23}) = 3, \quad \text{and} \quad (59)$$
$$1 \le C(e_{34}) \le W(e_{34}) = 3. \quad (60)$$

Because $C(e_{12}) = 2 \neq W(e_{12}) = 1$, there exists no feasible solution for $\varepsilon_{1:\{2\}}\varepsilon_{2:\{3\}}\varepsilon_{3:\{4\}}$.

3) The mathematical programming formulation for $\varepsilon_{1:\{2,3\}}\varepsilon_{2:\{3\}}\varepsilon_{3:\{4\}}$ is

$$C(e_{12}) + C(e_{13}) = 2, \quad (61)$$
$$C(e_{12}) = C(e_{23}), \quad (62)$$
$$C(e_{13}) + C(e_{23}) = C(e_{34}) = 2, \quad (63)$$

TABLE II
FINAL RESULTS OF EXAMPLES 1–3

Node I	Node subset I	items in $X_{1:I}$	the related 1-to-I 2-MP (e_{13},e_{12},e_{23},e_{34},e_{25})	the reliability for d=2 from node 1 to I
2	{2}	$\varepsilon_{1:\{2\}}$	$\varnothing$	0
3	{3}	$\varepsilon_{1:\{3\}}$	(2,0,0,0,0)	.8440
		$\varepsilon_{1:\{2\}}\varepsilon_{2:\{3\}}$	$\varnothing$	
		$\varepsilon_{1:\{2,3\}}\varepsilon_{2:\{3\}}$	(1,1,1,0,0)	
	{3,5}	$\varepsilon_{1:2}\varepsilon_{2:\{3,5\}}$	$\varnothing$	.6480
		$\varepsilon_{1:\{2,3\}}\varepsilon_{2:\{5\}}$	(1,1,0,0,1)	
		$\varepsilon_{1:\{2,3\}}\varepsilon_{2:\{3,5\}}$	$\varnothing$	
	{4}	$\varepsilon_{1:\{3\}}\varepsilon_{3:\{4\}}$	(2,0,0,2,0)	.5908
		$\varepsilon_{1:\{2\}}\varepsilon_{2:\{3\}}\varepsilon_{3:\{4\}}$	$\varnothing$	
		$\varepsilon_{1:\{2,3\}}\varepsilon_{2:\{3\}}\varepsilon_{3:\{4\}}$	(1,1,1,2,0)	
	{4,5}	$\varepsilon_{1:2}\varepsilon_{2:\{3,5\}}\varepsilon_{3:\{4\}}$	$\varnothing$	.5832
		$\varepsilon_{1:\{2,3\}}\varepsilon_{2:\{5\}}\varepsilon_{3:\{4\}}$	(1,1,0,1,1)	
		$\varepsilon_{1:\{2,3\}}\varepsilon_{2:\{3,5\}}\varepsilon_{3:\{4\}}$	$\varnothing$	
	{5}	$\varepsilon_{1:\{2\}}\varepsilon_{2:\{5\}}$	$\varnothing$	0

$$1 \le C(e_{12}) \le W(e_{12}) = 1, \tag{64}$$
$$1 \le C(e_{13}) \le W(e_{13}) = 2, \tag{65}$$
$$1 \le C(e_{23}) \le W(e_{23}) = 3, \quad \text{and} \tag{66}$$
$$1 \le C(e_{34}) \le W(e_{34}) = 3. \tag{67}$$

Therefore, $(C(e_{13}), C(e_{12}), C(e_{23}), C(e_{34}), C(e_{25})) = (1,1,1,2,0)$ is a 2-MP from nodes 1 to 4. □

There are several methods such as the disjoint subset method, the state-space decomposition method, and the inclusion-exclusion method to calculate the reliability [7]–[10]. Assume $X_1, X_2, \ldots, X_\delta$ are (one-to-many) d-MP, then the (one-to-many) reliability for level d using the inclusion-exclusion method is given by

$$\sum_{i=1}^{\delta} Pr(X_i) - \sum_{j=2}^{\delta}\sum_{i=1}^{j-1} Pr(X_i \cap X_j) + \sum_{j=3}^{\delta}\sum_{i=2}^{j-1}\sum_{k=1}^{i-1} Pr(X_i \cap X_j \cap X_k) + \ldots + (-1)^{\delta+1} Pr(X_1 \cap \ldots \cap X_\delta) \tag{68}$$

where $Pr\{X_{ij} = (x_{i1}, x_{i2}, \ldots, x_{im})\} = Pr\{X = (x^*_{i1}, x^*_{i2}, \ldots, x^*_{im}) | x_{ij} \le x^*_{ij} \text{ for } j = 1,2,\ldots,m\} = \prod_{j=1}^{m} Pr\{x_{ij} \le x^*_{ij}\}$.

The inclusion-exclusion method is applied next to find the reliability from node 1 to each node subset.

Example 3: Find all one-to-many reliabilities for $d = 2$ in terms of one-to-many 2-MP obtained in Example 2.

Solution: Also, only the procedure to find the reliability for $d = 2$ from nodes 1 to 4 is displayed as follow. Because $\{X_1 = (2,0,0,2,0), X_2 = (1,1,1,2,0)\}$ is the 2-MP set from nodes 1 to 4, the reliability for $d = 2$ from nodes 1 to 4 is

$$[Pr(X_1) + Pr(X_2)] - Pr\{X_1 \cap X_2\} \tag{69}$$
$$= [Pr(2,0,0,2,0) + Pr(1,1,1,2,0)] - Pr(2,1,1,2,0) \tag{70}$$
$$= [.7 \times 1 \times 1 \times .7 \times 1 + .9 \times .8 \times .9 \times 1 \times 1] - .7 \times .8 \times .9 \times .7 \times 1 \tag{71}$$
$$= .5908 \tag{72}$$

All one-to-many reliabilities are listed in the last column in Table II. □

VI. CONCLUSIONS

The UGFM had been proved very efficient in evaluating the one-to-many-targets AMNFDN reliability, which is a special MNFDN without any cycle. However, there is no UGFM for the one-to-many AMAFCN reliability problem. A new UGFM is introduced in this article using the simple operation of sets to calculate entire one-to-many AMAFCN reliabilities. It first searches for all (1:I)-MP for $I \subseteq V$ based on the proposed UGFM. Each (1:I)-MP is then transferred into the corresponding one-to-many d-MP. Finally, the entire one-to-many AMAFCN reliabilities are calculated based on one-to-many d-MP.

ACKNOWLEDGMENT

The author would like to thank the associate editor, and referees for their constructive comments, and recommendations, which have significantly improved the presentation of this paper.

REFERENCES

[1] W. J. Ke and S. D. Wang, "Reliability evaluation for distributed computing networks with imperfect nodes," *IEEE Trans. Reliability*, vol. 46, pp. 342–349, 1997.
[2] S. Soh and S. Rai, "An efficient cutset approach for evaluating communication-network reliability with heterogeneous link-capacities," *IEEE Trans. Reliability*, vol. 54, no. 1, pp. 133–144, 2005.
[3] K. S. Sohn, S. Y. Nam, and D. K. Sung, "A spare bandwidth sharing scheme based on network reliability," *IEEE Trans. Reliability*, vol. 54, no. 1, pp. 123–132, 2005.
[4] C. J. Colbourn, *The Combinatorics of Network Reliability*. New York: Oxford University Press, 1987.
[5] D. R. Shier, *Network Reliability and Algebraic Structure*. Oxford: Clarendon Press, 1991.
[6] Z. Tan, "Minimal cut sets of s-t networks with k-out-of-n nodes," *Reliability Engineering & System Safety*, vol. 82, no. 1, pp. 49–54, Oct. 2003.

[7] Y. K. Lin, "Study on the multicommodity reliability of a capacitated-flow network," *Computers & Mathematics with Applications*, vol. 42, pp. 255–264, 2001.
[8] Y. K. Lin, "A simple algorithm for reliability evaluation of a stochastic-flow network with node failure," *Computers & Operations Research*, vol. 28, pp. 1277–1285, 2001.
[9] Y. K. Lin, "Using minimal cuts to evaluate the system reliability of a stochastic-flow network with failures at nodes and arcs," *Reliability Engineering & System Safety*, vol. 75, pp. 41–46, 2002.
[10] Y. K. Lin, "Evaluate the performance of a stochastic-flow network with cost attribute in terms of minimal cuts," *Reliability Engineering & System Safety*, vol. 91, no. 5, pp. 539–545, 2006.
[11] J. M. Nahman, "MPs and cuts or networks exposed to common-cause failures," *IEEE Trans. Reliability*, vol. R-41, no. 84, pp. 76–80, 1992.
[12] Y. Shen, "A new simple algorithm for enumerating all minimal paths and cuts of a graph," *Microelectronics Reliability*, vol. 35, no. 6, pp. 973–976, 1995.
[13] W.-C. Yeh, "Multistate network reliability evaluation under maintenance cost constraint," *International Journal of Production Economics*, vol. 88, no. 1, pp. 73–83, 2004/3.
[14] W.-C. Yeh, "A new approach to evaluate the reliability of multistate networks under the cost constraint," *Omega: International Journal of Management Science*, vol. 33, no. 3, pp. 203–209, 2005/6.
[15] W.-C. Yeh, "A novel method for the network reliability in terms of capacitated-minimal-paths without knowing minimal-paths in advance," *Journal of the Operational Research Society*, vol. 56, no. 10, pp. 1235–1240, 2005/10.
[16] W.-C. Yeh, "Search for all d-mincuts of a limited-flow network," *Computers & Operations Research*, vol. 29, no. 13, pp. 1843–1858, 2002/6.
[17] W.-C. Yeh, "A new Monte Carlo method for the network reliability," in *Proceedings of the International Conference on Information Technology and Applications*, Bathurst, Australia, 2002.
[18] H. Boudali and J. B. Dugan, "A continuous-time Bayesian network reliability modeling, and analysis framework," *IEEE Trans. Reliability*, vol. 55, no. 1, pp. 86–97, 2006.
[19] I. Ushakov, "Universal generating function," *Sov. Journal of Comput. System Science*, vol. 24, no. 5, pp. 118–129, 1986.
[20] G. Levitin, *Universal Generating Function in Reliability Analysis and Optimization*. : Springer-Verlag, 2005.
[21] A. Lisnianski and G. Levitin, *Multi-State System Reliability. Assessment, Optimization and Applications*. : World Scientific, 2003.
[22] J. Malinowski and W. Preuss, "Reliability evaluation for tree-structured systems with multistate components," *Microelectronics Reliability*, vol. 36, pp. 9–17, 1996.
[23] G. Levitin, "Reliability evaluation for acyclic consecutively connected networks with multistate elements," *Reliability Engineering & System Safety*, vol. 73, no. 2, pp. 137–143, 2001.
[24] W.-C. Yeh, "Multistate-node acyclic networks reliability evaluation based on MC," *Reliability Engineering & System Safety*, vol. 81, no. 2, pp. 225–231, 2003.
[25] W.-C. Yeh, "The k-out-of-n acyclic multistate-node networks reliability evaluation using the universal generating function method," *Reliability Engineering & System Safety*, vol. 91, no. 7, pp. 800–808, 2006/7.

Wei-Chang Yeh is currently a Professor of the Department of Industrial Engineering and Management Engineering at National Tsing Hua University in Taiwan. He received his M.S., and Ph.D. from the Department of Industrial Engineering at the University of Texas at Arlington. His current research interests include network reliability theory, graph theory, deadlock problem, Linear Programming, and scheduling. He has published more than 90 papers in journals and conferences. He received the distinguished research award from 2004 to 2006 from the National Science Council in Taiwan. He is a member of the IEEE, and INFORMS.